EULERIAN-LAGRANGIAN MAPPING

FOR FINITE ELEMENT ANALYSES

Lars Olovsson

CROSS REFERENCE TO RELATED APPLICATIONS

This application is related to U.S. Patent Application No. 09/836,490 filed April 17, 2001 entitled "Implicit-Explicit Switching for Finite Element Analysis", which is hereby incorporated by reference in its entirety.

BACKGROUND OF THE INVENTION

1. Field of the invention

The present invention relates generally to computer simulations using the finite element method, and more specifically, to a method for switching from an explicit Eulerian formulation to an implicit Lagrangian formulation in a finite element simulation of motion.

2. Related art

The finite element method is a technique for obtaining approximate numerical solutions to boundary value problems which predict the response of physical systems subjected to external loads. The finite element method is described in detail by Thomas J. R. Hughes in "The Finite Element Method" (1987), published by Prentice-Hall, Inc., New Jersey, which is incorporated herein by this reference in its entirety. One common use of the finite element method is in the field of solid mechanics where it is used to analyze structural problems such as the formation of stamped sheet metal parts or the springback of stamped

sheet metal parts. Another common application is injection molding. The equations

describing the physical event of interest are generally overly complex to be solved exactly.

The finite element method is a technique where the geometry of the analysed structure

is approximated as a set of points in space. The points, which are referred to as nodes, are

5   connected together to form finite elements. This process is referred to as discretization.

Figure 1 shows the discretization of a wrench. The discretization of the geometry allows the

differential equations to be approximated as finite sized matrix equations.

The finite element method can be used to run two or three dimensional simulations.

In a two-dimensional (2D) simulation the elements are areas. In a three-dimensional (3D)

10   simulation the elements are volumes. All of the simulations illustrated in this example are 3D

simulations. The elements are therefore three dimensional volumes. However, for ease of

illustration and explanation cross sections are used to illustrate the invention. The fill

fraction, which in these 3D simulations is referred to as the volume fraction, would be an area

fraction in the case of a 2D simulation. The elements and nodes form a mesh or grid, and

15   these terms are used interchangeably throughout this application. Additionally, the elements

are shown as cubes or rectangles, however other geometric shapes may be used.

In structural mechanics, the matrix equations describe the relationship between the

stress and velocity fields and the acceleration field at a specific instant in time. To follow the

deformation process, one needs to integrate the matrix equations in time. Due to non-

20   linearities, an exact integration is generally not possible. A time discretization is necessary

and one usually relies on a finite difference scheme to drive the solution forward in time. The

matrix equations may be explicitly or implicitly integrated. Further explanation of explicit

and implicit integration can be found in U.S. Patent Application No. 09/836,490 filed April

17, 2001 entitled "Implicit-Explicit Switching for Finite Element Analysis", which is hereby

25   incorporated by reference in its entirety.

2

The finite element method has been implemented in finite element software, such as LS-DYNA, a commercial finite element analysis program developed by Livermore Software Technology Corporation of Livermore, California.

I.   Explicit Eulerian Material and Void Formulation

5   An *Eulerian* finite element formulation is a method where the element grid is fixed in space. The material flows through the grid of elements. The *material and void* formulation refers to a special method where the elements are not necessarily filled with material. Each element contains a certain volume fraction(3D) or area fraction (2D) of material, ranging from 0 to 1.

10   *Explicit* time integration is suitable in situations where there are large non-linearities. No equation systems need to be solved and each time increment is computationally inexpensive. Small time increments are required in order to avoid numerical instabilities and diverging solutions. Explicit time integration techniques are well known in the art.

The sequence of pictures in Figure 2 are snapshots from an Eulerian material and void simulation. It is a model of a high-speed impact where a piece of falling steel hits a rigid wall. The dark area indicates the location of the material. The steel plastically deforms upon impact. The first box shows the steel block just before impact, the middle box during impact, and the third box after impact has occurred. Observe that the element grid is fixed in space as the steel material flows.

20   An advantage of the Eulerian formulation is that there are no element distortions. Distorted elements are bad for the numerical accuracy and can even lead to crashing simulations.

A disadvantage is that the flux of material between elements (fixed in space) slows down the simulation and is bad for numerical accuracy. Another disadvantage is that the

25   need for stability in the explicit time integration scheme puts a limit on the maximum allowed

3

time step size. Therefore, slow and quasi-static processes, such as springback, are better

treated with implicit methods.

II.   Implicit Lagrangian Formulation

5   A *Lagrangian* Finite Element formulation is a method where the element grid follows

the material flow. That is, each element represents the same piece of material throughout the

complete simulation. Figure 3 shows how the Lagrangian element grid follows the material

flow.

Implicit time integration is suitable in static, quasi-static and in certain slow, dynamic

10   simulations. An equation system needs to be solved for each time increment. Implicit time

integration techniques are well known in the art.

An advantage of the Lagrangian formulation is that the numerical implementation is

relatively straightforward. The numerical accuracy is good at small deformations. Another

advantage is that certain implicit time integration schemes provide an unconditional stability.

15   Therefore, large time steps can be allowed in quasi-static simulations.

A disadvantage of the Lagrangian formulation is that element distortions cause

numerical problems when dealing with large deformations. Further, solving complex

equation systems makes each time step rather costly in terms of time and computing

resources.

20   SUMMARY OF THE INVENTION

The present invention includes a method that overcomes the disadvantages of

conventional finite element simulations. The method accomplishes this by switching from an

Eulerian to a Lagrangian formulation during a finite element simulation. The material and

void based Eulerian formulation is suitable for the study of mechanical processes involving

4

large deformations. An implicit Lagrangian formulation is preferable for the treatment of quasi-static and small deformation processes.

Some events are better described with a mix of the two formulations. For example, many forging and injection molding processes involve large deformations requiring an

5 Eulerian formulation. However, a subsequent cooling down and springback process is better treated with an implicit Lagrangian formulation because the springback is a relatively small deformation in comparison with the initial forging or injection molding process. This can be realized by first simulating the forging or molding operation with an Eulerian simulation. Then, the resulting Eulerian mesh is mapped onto a new Lagrangian mesh and the springback

10 and cooling down process is analyzed with an implicit Lagrangian formulation.

In one embodiment of the present invention, a method for performing a finite element simulation is disclosed. The method includes switching from an Eulerian formulation to a Lagrangian formulation during a finite element simulation.

In another embodiment of the present invention, a computer readable storage medium

15 storing one or more computer programs for performing a finite element simulation is disclosed. The computer programs include instructions for automatically switching between an Eulerian formulation and a Lagrangian formulation during the finite element simulation.

In another embodiment of the present invention, a computer system is disclosed. The computer system includes one or more computers and one or more computer programs

20 running on the computers where the computer programs are for performing a finite element simulation. The computer programs include computer instructions for automatically switching between an Eulerian formulation and a Lagrangian formulation during the finite element simulation.

In another embodiment of the present invention, a data signal embodied in a carrier

25 wave is disclosed. The data signal includes one or more computer programs for performing a

finite element simulation. The computer programs include instructions for automatically switching between an Eulerian formulation and a Lagrangian formulation during the finite element simulation.

Other embodiments, aspects, and advantages of the present invention will become

5 apparent from the following descriptions, the accompanying drawings, and the accompanying claims.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and for further features

10 and advantages, reference is now made to the following description taken in conjunction with the accompanying drawings, in which:

FIG. 1 illustrates the exact geometry and finite element discretization of a wrench.

FIG. 2 illustrates an Eulerian simulation of a high speed impact of a steel object impacting a rigid wall and the resultant deformation.

15 FIG. 3 illustrates a Lagrangian simulation of a high speed impact of a steel object impacting a rigid wall and the resultant deformation.

FIG. 4 is a block diagram of an exemplary computer system upon which the methods of the present invention can be performed, according to an embodiment of the present invention.

20 FIG. 5 is a flowchart of an exemplary explicit Eulerian simulation and mapping of a new mesh according to the first phase of an embodiment of the present invention.

FIG. 6 is an exemplary illustration of an explicit Eulerian time step E3 of FIG. 5 broken down into individual steps.

FIG. 7 is an exemplary illustration of steps E6 - E8 of FIG. 5.

FIG. 8 is a flowchart of an exemplary implicit Lagrangian simulation according to a second phase of an embodiment of the present invention.

FIG. 9 illustrates a simulation of a box impacting a beam.

FIGS. 10a-10c illustrate the process of orthogonal projection of nodes to the material surface according to an embodiment of the present invention.

FIG. 11 illustrates the beam of FIGS. 9 and 10a-10c after removing empty cells according to an embodiment of the present invention.

FIG. 12 illustrates the beam of FIGS 9-11 before and after springback.

FIG. 13a-d illustrate a simulation of the forging process of a cup utilizing the present invention.

DETAILED DESCRIPTION OF THE INVENTION

The preferred embodiments of the present invention and their advantages are best understood by referring to FIGS. 1 through 5 of the drawings. Like numerals are used for like and corresponding parts of the various drawings.

System Overview

FIG. 4 is a block diagram of an exemplary computer system 2 upon which the methods of the present invention can be performed, according to an embodiment of the present invention. Computer system 2 can include one or more suitable processing facilities such as microcomputers/workstations, minicomputers, mainframes, and/or massively parallel processing computers running suitable operating systems. Examples of suitable microcomputers/workstations include the IBM PC from IBM Corporation of Armonk, N.Y., and the Sparc-Station from Sun Microsystems Corporation of Mountain View, Calif. Examples of suitable minicomputers are the VAX 750 from Digital Equipment Corporation of Maynard, Mass. and the AS/400 from IBM. An example of a suitable mainframe is the

System-390 series from IBM. Examples of suitable massively parallel processing computers

include the Cray T3E or the Cray T90 from Cray Corporation of Mendota Heights, Minn.

Examples of suitable operating systems include any of the various versions of Unix, VMS

from Digital Equipment, or OS/2 from IBM. Computer systems used in the development of

5   the invention include the IBM RS-6000, the Hewlett-Packard V-Class, and the Silicon

Graphics Origin 2000, all of which are engineering workstations or servers running under the

UNIX operating system, as ported by each hardware vendor. Those skilled in the art will

recognize that other types of suitable computer systems and suitable operating systems can be

used in accordance with the present invention.

10   Computer system 2 includes the following components: a central processing unit

(CPU) 4, a random access memory (RAM) 6, a read-only memory (ROM) 8, a storage device

10, an input/output (I/O) device 12, display 14, and network connection 16. Although only a

single instance of each component is shown, it should be recognized that computer system 2

can include one or more of any of the components shown in FIG. 4. Computer system 2 can

15   also include other suitable computer components. For clarity, interconnections between the

various components of computer system 2 are not shown. Additionally, multiple computers 2

can be connected via network connection 16, either through a local area network (LAN) or

through a larger network such as the internet. A computer "farm" may be configured wherein

each computer system 2 or CPU 4 calculates a portion of the simulation. The portions are

20   then later integrated to arrive at the complete simulation. In some cases, this networked

"farm" may be more efficient than using one more powerful computer such as a workstation

or mainframe.

         CPU 4 can be any type of processor, such as a Pentium processor from Intel

Corporation or an Alpha EV5 from Digital Equipment Corporation, used to process

25   information in computer system 2.

RAM 6 can be any type of memory device, including dynamic and/or static RAM, used to store any number of different types of information required by computer system 2. For example, RAM 6 can store computer programs for performing the method of the present invention.

5   ROM 8 can be any type of read-only memory device, including PROMs, EPROMs and EEPROMs, used to store any number of different types of information required by computer system 2. For example, ROM 8 can store information concerning initialization routines or even some or all of the computer programs for performing the method of the present invention.

10   Storage device 10 can be any type of mass storage device, including magnetic, electrical, and optical-type storage devices, used to store any number of different types of information required by computer system 2. For example, storage device 10 can store various types of data or even some or all of the computer programs for performing the method of the present invention. The computer programs may be transmitted over network

15   connection 16 as a data signal.

I/O device 12 can be any type of input/output device, including keyboards, printers, or any other type of device, by which information can be entered into or received from computer system 2.

Display 14 can be any type of display device, including those of the CRT and flat-

20   panel display type, used to display information. For example, display 14 can display the results of a computer simulation, which results from performing the methods of the present invention, via a graphical user interface.

Eulerian and Lagrangian Phases

A finite element analysis according to an embodiment of the invention is split into

25   two phases. The Eularian and Lagrangian phase can be implicit or explicit. For illustrative

purposes an explicit Eularian phase and an implicit Lagrangian phase are described. The

explicit Eulerian phase is described by the flow diagram in Figure 5 and the implicit

Lagrangian phase is described by the flow diagram in Figure 8.

There are two main steps in the switching from the Eulerian material and void

5   formulation to the Lagrangian formulation. First, a new mesh is automatically created, and

then the results are mapped from the Eulerian mesh onto a Lagrangian mesh.

Explicit Eulerian Phase

Referring to the flow chart of Figure 5, the explicit Eulerian phase is described.

First, in step E1, the simulation program (LS-DYNA) is started. Then in step E2,

10   information describing the process to be analysed is read from the storage device 10 of

computer system 2. A suitable time step size, dt, is computed and the current time is set to

$t=dt$.

In step E3 an explicit Eulerian time step is taken. Step three is illustrated in further

detail in Figure 6. In step E3a, the velocity field and the geometry at the current time, t, are

15   computed. Note that the nodes follow the motion of the material (mesh configuration $M1$).

In step E3b, the incremental deformation of the structure is computed and the current state of

the material is estimated. In steps E3c-d, the nodes are moved back to their initial position

(mesh configuration M0), and the solution is mapped from mesh configuration M1 to mesh

configuration M0. This process is well known in the art.

20   Referring once again to Fig. 5, in step E4, the system will check if the termination

time, $tmax$, has been reached. If it has been reached, the process will continue on to step E6.

If it has not been reached, the system will update the time, t, for a new time step increment in

step E5, and continue on until time $tmax$ has been reached. The time $tmax$ is a the overall

time allotted for a simulation or part of a simulation, and is user selectable. In this instance, it

25   is the time at which the Eularian portion of the simulation will terminate, i.e., when no more

Eularian time steps will be performed. When the user sets up the simulation, he will decide at which point in time he wishes to conclude the Eularian portion of the simulation. Alternatively this may be preset in the software for certain simulations.

5   Steps E6-E8 are illustrated in further detail in Fig. 7. In step E6, the system will sort out partially filled elements that are cut by the material surface. Figure 7 illustrates a two dimensional cross section of a material. The material is represented by the dark object. The outer boundary, as seen in the cross section of the material, is referred to as the surface of the material in this 3D simulation. The person performing the simulation will decide what cross section and thus boundary to analyze in a two dimensional simulation or what surface to
10   analyze in a three dimensional simulation. Some nodes of partially filled elements are projected orthogonally down to the material surface, and some nodes are projected up to the material surface. Partially filled elements are treated differently depending on their fill fraction, η.

     The threshold fill fraction η will be used to determine which nodes of an element are
15   mapped to the surface. This value can range from about 0.5 to 1.0 and is preferably 0.7. An obvious transition between outwards and inwards projection of nodes would be at $\eta=0.5$. However, the inwards projection creates smaller elements and potentially troublesome element distortions. A transition at $\eta=0.7$ has proven to generate more well shaped elements, which is important for the condition number of the subsequently formed stiffness matrix.
20   The new mesh configuration is referred to as *M2*.

     If the fill fraction η is greater than 0.7, then nodes outside the material domain are projected orthogonally down to the material surface. Nodes inside the material domain are not moved.

11

If the fill fraction η is less than or equal to 0.7, then nodes inside the material domain are projected orthogonally outwards to the material surface. Nodes outside the material domain are not moved.

In step E7 the standard Eulerian mapping algorithm is used to map the solution from configuration *M0* to configuration *M2* i.e. the nodes are connected to form the mesh. Frequently applied mapping methods are the Donor Cell and the van Leer algorithms, which are well known in the art.

In step E8 any empty elements, that is, elements with a fill fraction η=0, are deleted from the grid.

Referring once again to Fig. 5, in step E9, closely spaced nodes are merged. Different nodes might be projected to almost the same point on the material surface, and some brick (orthogonal Eularian grid) elements degenerate to 4, 5, 6 or 7 node elements. Nodes located a small distance from each other are merged according to the following formula:

$$d < 0.2\sqrt[3]{V} \Rightarrow \text{merge nodes},$$

where *d* is the distance between two adjacent nodes in one element and *V* is the element volume.

In step E10 information about the current mesh, the material state and the velocity field are saved to the file *dynain* on the storage device 10 of the system 2 shown in Figure 4. This information will be used during initialization of the subsequent implicit Lagrangian simulation phase.

Finally in step E11, the Eulerian phase of the simulation is terminated.

Implicit Lagrangian Phase

Referring to Fig. 8, the implicit Lagrangian phase of the simulation program is started in step I1. In step I2, information describing the process to be analysed is read from the

12

storage device 10. Initial stresses and the geometry are defined by the data in the file *dynain*. The time step size, *dt2*, is user defined and the current time is set to *t=dt2*. The variable indicating the equilibrium iteration number is set to *iter=1*.

5   In step I3 an implicit Lagrangian time step is taken. The deformation and velocity fields at time *t* are predicted. The material state, based upon the predicted deformation, is estimated. The predicted state at the iteration is denoted as S_{iter}.

In step I4 the system determines if equilibrium is satisfied for the predicted deformation and material state, i.e., if there is convergence of the mathematical solution.

If too far away from equilibrium, another iteration will be prepared as seen in step I5

10   (set *iter:=iter+1)* and another implicit Lagrangian time step will be taken in step I3 until convergence has been achieved.

Once convergence has been achieved, the system will proceed to step I6. In step I6 the system checks if the termination time, *tmax2*, has been reached. If *tmax2* has not been reached the system will update the time, *t*, in step I7, for a new time step increment, set the

15   iteration value to 1 (set *iter=1)*, and return to step I3 where it will perform another implicit Lagrangian time step. The time *tmax2* is the maximum time for the Lagrangian portion of the simulation, and is selectable by the user when configuring the simulation. Generally it would be set to a time when adequate springback has already occurred. Alternatively, *tmax2* can be selected by the system. If *tmax2* has been reached, the simulation will be stopped in step I8.

20   Illustrative Example

A simple application of the invention is presented below in Figs. 9-11 to illustrate operation of the invention. A box impacts a cantilever beam, which is subsequently elastically deformed and returns back to its initial state. This example is used to illustrate the concepts of the invention in a simple application. In this elastic deformation, the springback

25   is in theory equal to the initial deformation. However, the method of the present invention is

generally tailored for applications such as metal stamping or injection molding, where the

initial deformation is substantially plastic and quite large in comparison to the subsequent

springback. In these applications, the final dimensions of the finished product after

springback are critical to the manufacturer and end user. It is in these cases that using a

5   combination of the Eulerian and Lagrangian methods is most advantageous, and leads to a

more accurate prediction of dimensions than prior systems.

The impact event is simulated dynamically with an explicit time integration for a

short period of time, as seen in Figure 9. The first frame of Figure 9 captures the instant the

falling box makes contact with the beam (time t=0). In the second and third frames, the beam

10   has been deflected by the impact. The third frame shows the maximum deflection of the

beam.

At the maximum deflection point, which is determined by the user, a Lagrangian

model of the beam is created and a quasi-static implicit springback simulation is performed.

Different steps of the process are presented in Figures 10a-c and 11.

15   Figure 10a shows the beam at its maximum deflection and a box around the tip of the

beam which is enlarged in Figures 10b and 10c. Figure 10b shows the tip of the beam before

projection of the nodes to the surface of the beam. As discussed above the nodes outside the

surface of the beam will either be projected down to the surface or the nodes inside the beam

will be projected out the surface of the beam depending on the fill fraction η of the associated

20   element. Figure 10c shows the tip of the beam after the nodes have been projected to the

surface of the beam. Figure 11 shows the beam after removing the empty cells, i.e., those

cells having a fill fraction $\eta=0$.

Figure 12 shows the subsequent implicit springback analysis. On the left, Figure 12

shows the beam before springback including the fringe of Von Misses stresses. On the right,

25   Figure 12 shows the beam after a simulated springback using the newly mapped grid and a

Lagrangian analysis according to the invention as described previously. Note that the beam is not completely straight as a result of deficiencies in the previously described mapping process at the material boundaries. These deficiencies are minimized by minimizing the movement of the nodes and elements. Only elements in the vicinity of the material boundary

5   are moved when creating the Lagrangian mesh. This is important for the numerical accuracy in the process of mapping the solution from the Eulerian mesh onto the Lagrangian one. In fact, in the interior of the body, where no elements are moved, the mapping will be exact and no information will be lost.

If the Eulerian and Lagrangian simulations were perfectly accurate, in this example of

10   elastic deformation the beam would be precisely as it was before any deformation. In this example, the initial deformation was designed to be completely elastic and thus the springback was of an equal magnitude to the deformation. However, where the initial plastic deformation is of relatively large magnitude when compared to the subsequent springback, as is the case in such applications as stamping and molding, using the Eulerian formulation for

15   the large deformation and the implicit Lagrangian formulation for the springback results in a more tailored and accurate simulation than is otherwise possible with prior solutions.

Figures 13a-13d illustrate another application of the invention.

Referring to Figure 13a, a thick cylindrical disk of material is to be formed into a cylindrical cup using a forging process. Figure 13a shows the disk material 300 inside a

20   larger computational domain 305 which is initially void. The outline of the forging tool 310 is shown in each figure 13a - 13d. The bottom surface 310a of the forging tool 310, which may be the bottom of a piston in some applications moves down to force the material up through the side cavity 310b of the forging tool 310. Only one quarter of this axisymmetric model is shown. In this Eularian phase of the simulation note that the grid and its elements

25   are fixed and the material moves through the fixed grid.

As the forging begins, the bottom surface 310a advances downward, causing the disk material 300 to extrude upward into the cavity 310b between the sides of tool 310. This extruded disk material elongates as the tool advances, forming the sides of the cylindrical cup. This progression can be seen from Figure 13a to Figure 13b.

5   In Figure 13c, after the forging finishes, the nodes are mapped onto the boundary or surface of the disk material (the cup).

Figure 13d illustrates the cup formed from disk material 300. The final dimensions of the cup are automatically and accurately simulated with the present invention. The internal stresses from the forming process shown in Figures 13a-b are used to determine the

10   subsequent springback of the material, so that the final and accurate dimensions of the cup are simulated. The present invention performs all of this automatically, thus providing the engineer running the simulation a powerful, effective, and very convenient tool.

Different numerical solution procedures are preferable in different situations. The modeling of certain mechanical processes can not be treated efficiently with one solution

15   procedure solely. The invention allows a switching between methods and a wider range of mechanical processes can now be analysed. The automatic Eulerian - Lagrangian switching method is a robust automatic procedure that can obtain solutions which are very accurate and can be obtained with minimal operator interaction. Conventional simulation techniques often could not produce an accurate solution and required significant operator interaction.

20   While particular embodiments of the present invention and their advantages have been shown and described, it should be understood that various changes, substitutions, and alterations can be made therein without departing from the spirit and scope of the invention as defined by the appended claims. For example, although it has been described how the present invention can be used in formation simulations and springback simulations, those

25   skilled in the art will recognize that the present invention can be applied to a wide variety of

16

problems relating to non-linear static or dynamic analysis using the finite element method, such as general structural mechanics and stress analysis, fluid mechanics, biomechanics, heat transfer, geomechanics, aeromechanics, chemical reactions, acoustics, and electric and magnetic fields. In addition, an explicit Eulerian and an implicit Lagrangian method have

5   been illustrated; however, the Eulerian and Lagrangian methods can be either implicit or explicit.